Beyond Stemming and Lemmatization: Ultra-stemming to Improve Automatic Text Summarization

Juan-Manuel Torres-Moreno[1,2]

[1] Laboratoire Informatique d'Avignon,
BP 91228 84911, Avignon, Cedex 09, France
juan-manuel.torres@univ-avignon.fr
[2] École Polytechnique de Montréal,
CP. 6128 succursale Centre-ville, Montréal, Québec, Canada

Abstract

In Automatic Text Summarization, preprocessing is an important phase to reduce the space of textual representation. Classically, stemming and lemmatization have been widely used for normalizing words. However, even using normalization on large texts, the curse of dimensionality can disturb the performance of summarizers. This paper describes a new method for normalization of words to further reduce the space of representation. We propose to reduce each word to its initial letters, as a form of Ultra-stemming. The results show that Ultra-stemming not only preserve the content of summaries produced by this representation, but often the performances of the systems can be dramatically improved. Summaries on trilingual corpora were evaluated automatically with Fresa. Results confirm an increase in the performance, regardless of summarizer system used.

Keywords: Automatic Text Summarization, Lemmatization, Stemming, Ultra-Stemming

1 Introduction

In Natural Language Processing (NLP), pre-processing aims to reduce the complexity of the vocabulary of the documents. Pre-processing eliminates the punctuation, filters the function words and normalizes the morphological variants. In particular, the lemmatization and stemming are two commonly used techniques to normalize morphological variants.

The lexeme or word-root is the part that does not change and contains its meaning. The morpheme or variable part is added to the lexeme to form new words. Morphological analysis is a very important phase of pre-processing of NLP systems because it allows to reduce the dimension of the vector space representation in systems of Information Retrieval [3, 32]. Several applications such as Automatic Summarization, Document Indexing, Textual Classification and Question-Answering systems among others[3], utilize this reduction. However, the realization of this analysis may require the use of external resources (dictionaries, parsers, rules, etc.) which can be expensive and difficult to build, depending on language or specific domain [32]. Some algorithms are capable to detect statistically morphological families (posed as a classification problem), avoiding the utilization of external resources or a priori knowledge of a language.

Automatic Text Summarization (ATS) is the process to automatically generate a compressed version of a source document [41]. Query-oriented summaries focus on a user's request, and extract the information related to the specified topic given explicitly in the form of a query [11]. Generic mono-document summarization tries to cover as much as possible the information content. Multi-document summarization is a task oriented to creating a summary from

a heterogeneous set of documents on a focused topic. Over the past years, extensive experiments on query-oriented multi-document summarization have been carried out. Extractive Summarization produces summaries choosing a subset of representative sentences from original documents. Sentences are ordered, then assembled according to their relevance to generate the final summary [31].

This article introduces a new method of normalization of words that reduces the textual representation space, in order to improve the efficiency of Automatic Text Summarizers based on Vector Space Model (VSM). We propose Ultra-stemming which reduces every word(s) to its initial(s) letter(s). Results show that Ultra-stemming not only preserves the content of the summaries generated using this new representation, but often, surprisingly the performance can be dramatically improved. To our knowledge, in summary tasks no automatic stemming method has explored this extreme possibility. Ultra-stemming could be an interesting alternative for ATS of documents in languages π, where electronic linguistic resources are rare. In these languages, there are a notable absence of lemmatizers, stemmers, parsers, dictionaries, corpora and language resources in general (such as Nahuatl and other American Indian languages). Our tests on trilingual corpora evaluated by the FRESA algorithm confirm the increase of performance regardless of summarizer used and a big reduction of complexity in space and time required to generate summaries. Related work is given in Section 2. Section 3 presents our Ultra-stemming strategies coupled with methods of Automatic Text Summarization. Experiments are presented in Section 5, followed by a discussion and the conclusions in Section 6.

2 Related works

There are several morphological analysis methods [20, 21]. Examples of these algorithms are the Comparison of Graphs [19], the use of n-grams [16, 32], the search for analogies [27], the surface models based on rules [25, 37], the probabilistic models [10], the segmentation by optimization [9, 18], the unsupervised learning of morphological families by ascending hierarchical classification [4], the lemmatization using Levenshtein distances [12] or identifying suffixes through entropy [44]. These methods are distinguished by the type of results obtained, by the identification of lemmas, stems or suffixes. FLEMM[1] [14] is an analyzer for French which requires a text previously labeled by WINBRILL[2] or by TREETAGGER[3]. FLEMM produces, among other results, the lemma of each word of the input text. TREETAGGER [22] is a multilingual tool that allows to annotate texts with information of *Parts-Of-Speech* (POS) [4] and with information of lemmatization. TREETAGGER uses supervised machine learning and probabilistic methods [7, 38]. It can be adapted to other languages as long as the lexical resources and manually labeled corpora are available. FREELING is another example of a popular multilingual lemmatizer[5].

Stemming transforms the variants of words into truncated forms. Two popular stemming algorithms are the Porter stemming algorithm [37] and the Paice algorithm [35]. The methods of stemming and lemmatization can be applied when the terms are morphologically similar. Otherwise when the similarity is semantic, lexical search methods must be used. To reduce semantic variation, some systems use long dictionaries. Another systems use thesauri to associate words to entirely different morphological forms [36]. Both methods are complementary since

[1] FLEMM is available in web site: `http://www.univ-nancy2.fr/pers/namer/Telecharger_Flemm.htm`

[2] WINBRILL is available in web site: `http://www.atilf.fr/scripts/mep.exe?HTML=mep_winbrill.txt;OUVRIR_MENU=1`

[3] TREETAGGER is available in web site: `http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html`

[4] The types of words are, for example, nouns, verbs, infinitives and particles.

[5] FREELING is available in web site: `http://www.lsi.upc.edu/~nlp/freeling/`

the stemming verifies similarities in the spelling level to infer lexical proximity, while the lexical algorithms use terminographic data with links to synonyms. [24]. [17] presents an unsupervised genetic algorithm for stemming inflectional languages. [46] proposes using morphological merged families into a single term to reduce the linguistic variety of Spanish indexed texts.

Lexematization [34] seeks morphological rearrangement of words belonging to the same family using automatic acquisition of morphological knowledge directly from the texts. Although the constitution on morphological families may be interesting in itself, its main interest lies in the benefits it produces for use as normalization mechanism (instead or in addition to stemming or lemmatization) in specific application domains. Probably the most common application domain is indexing terms in systems of Information Retrieval (IR). In recent years there have been numerous articles analyzing in different languages the efficiency of stemming/lemmatization in IR. In addition, significant progress has been made in IR in European languages other than English. In particular, [23] have evaluated corpora of CLEF evaluation campaigns [6] (eight European languages). Their results show that morphological normalization techniques increase the efficiency of the IR systems and it can be used independently of the language. Reduction algorithms using syntactic and morphosyntactic variations have shown a significant reduction of storage costs and management by storing lexemes rather than terms [45]. [1] works on the impacts of compound words and standardization in IR, finding no significant performance differences between stemming and lemmatization.

However, the reality is that the linguistic resources necessary to establish morphological relationships without pre-defined rules are not available for all languages and all domains, without mention the constant creation of neologisms [8]. The proposed solution for the specific task of automatic summarization is the Ultra-stemming of letters.

Research in ATS was introduced by H.P. Luhn in 1958 [30]. In the strategy proposed by Luhn, the sentences are scored for their component word values as determined by tf*idf-like weights. Scored sentences are then ranked and selected from the top until some summary length threshold is reached. Finally, the summary is generated by assembling the selected sentences in original source order. Although fairly simple, this extractive methodology is still used in current approaches. Later on, [13] extended this work by adding simple heuristic features of sentences such as their position in the text or some key phrases indicating the importance of the sentences. As the range of possible features for source characterization widened, choosing appropriate features, feature weights and feature combinations have became a central issue. A natural way to tackle this problem is to consider sentence extraction as a classification task. To this end, several machine learning approaches that uses document-summary pairs have been proposed [26, 40].

3 Pre-processing and Ultra-stemming

The following subsections present formally the details of the corpora studied and the proposed text pre-processing method.

3.1 Summarization Corpora Description

To study the impact of Ultra-stemming in automatic summary tasks, we used corpora in three languages: English, Spanish and French. The corpora are heterogeneous, and different tasks are representive of Automatic Summarization: generic multi-document summary and mono-document guided by a subject.

[6] Cross-Language Evaluation Forum, `http://www.clef-campaign.org/`

- Corpus in English. Piloted by NIST in Document Understanding Conference[7] (DUC) the Task 2 of DUC'04[8], aims to produce a short summary of a cluster of related documents. We studied generic multi-document-summarization in English using data from DUC'04. This corpus with 300K words is compound of 50 clusters, 10 documents each.
- Corpus in Spanish. Generic single-document summarization using a corpus from the journal *Medicina Clínica*[9], which is composed of 50 medical articles in Spanish, each one with its corresponding author abstract. This corpus contains 125K words.
- Corpus in French. We have studied generic single-document summarization using the Canadian French Sociological Articles corpus, generated from the journal *Perspectives interdisciplinaires sur le travail et la santé* (PISTES)[10]. It contains 50 sociological articles in French, each one with its corresponding author abstract. This corpus contains near 400K words.

Table 1 presents the basic statistics on tokens, types and characters of the three summarization corpora studied.

Corpus	**Language**	**Tokens**	**Types**	**Letters**
DUC'04	English	294 236	17 780	1 834 167
MEDICINA CLÍNICA	Spanish	125 024	9 657	793 937
PISTES	French	380 992	18 887	2 590 623

Table 1: Basic Statistics for the three Summarization corpora.

Additionally, three large and heterogeneous corpora (generated from novels, newspaper articles and news on the Internet) were created to measure statistics of each language. These corpora contains several million tokens in English, Spanish and French. Table 2 presents basic statistics on tokens and characters of the three generic corpora.

Generic Corpus	**Tokens**	**Letters**
ENGLISH	29 346 289	177 717 720
SPANISH	21 445 694	134 461 092
FRENCH	17 734 663	111 169 782

Table 2: Basic Statistics for the three Language Generic corpora.

3.2 Ultra-stemming

The first step to represent documents in a suitable space is the pre-processing. As we use extractive summarization as task, documents have to be chunked into cohesive textual segments that will be assembled to produce the summary. Pre-processing is very important because the selection of segments is based on words or bigrams of words. The choice was made to split documents into full sentences, in this way obtaining textual segments that are likely to be grammatically correct. Afterwards, sentences pass through several basic normalization steps in order to reduce computational complexity. An example of document pre-processing is given in Table 3. The process is composed by the following steps:

[7] http://duc.nist.gov
[8] http://www-nlpir.nist.gov/projects/duc/guidelines/2004.html
[9] http://www.elsevier.es/revistas/ctl_servlet?_f=7032&revistaid=2
[10] http://www.pistes.uqam.ca/

1. **Sentence splitting**: a simple rule-based method is used for sentence splitting. Documents are chunked at the dot, exclamation and question mark signs.
2. **Sentence filtering**: words are converted to lowercase and cleared up from sloppy punctuation. Words with less than 2 occurrences ($f < 2$) are eliminated (*Hapax legomenon* presents once in a document). Words that do not carry meaning such as functional or very common words are removed. Small stop-lists (depending of language) are used in this step.
3. **Word normalization**: remaining words are replaced by their canonical form using lemmatization, stemming, Ultra-stemming or none of them (raw text).
4. **Text Vectorization**: Documents are vectorized in a matrix $S_{[P \times N]}$ of P sentences and N columns, that represent the occurrences of a letter (Ultra-stemming) or a word (Lemmatization/Stemming/Raw) j, $j = 1, 2, ..., N$ in the sentence i, $i = 1, 2, ..., P$.
5. **Summary generation**: each summary is generated by a summarizer based on VSM.

For Ultra-stemming using $n = 1$ (FIX_1), the maximum dimension N may be up to 32 letters. This generates very compact and efficient matrices, as discussed in 3.4.

Original	A federal judge Monday found President Clinton in civil contempt of court for lying in a deposition about the nature of his sexual relationship with former White House intern Monica S. Lewinsky. Clinton, in a January 1998 deposition in the Paula Jones sexual harassment case, swore that he did not have a sexual relationship with Lewinsky. Clinton later explained that he did not believe he had lied in the case because the type of sex he had with Lewinsky did not fall under the definition of sexual relations used in the case.
Splitted	s0/A federal judge Monday found President Clinton in civil contempt of court for lying in a deposition about the nature of his sexual relationship with former White House intern Monica S. Lewinsky. s1/Clinton, in a January 1998 deposition in the Paula Jones sexual harassment case, swore that he did not have a sexual relationship with Lewinsky. s2/Clinton later explained that he did not believe he had lied in the case because the type of sex he had with Lewinsky did not fall under the definition of sexual relations used in the case.
Stemming	s0/feder judg monday found presid clinton civil contempt court lying in deposit natur sexual relationship former white hous intern monica lewinski s1/clinton januari deposit paula jone sexual harass case swore sexual relationship lewinski s2/clinton explain believ lie case type sex lewinski fall denit sexual relat case
FIX_1	s0/f j m f p c c c c l d n s r f w h i m l s1/c j d p j s h c s s r l s2/c l e b l c t s l f d s r u c
Matrix	letter:c d e f h i j l m n p r s u w s0: 4 0 0 0 1 1 1 2 2 1 1 1 1 0 1 s1: 2 1 0 0 1 0 2 1 0 0 1 1 2 0 0 s2: 3 1 1 0 0 0 0 3 0 0 0 1 2 1 0

Table 3: Example of some pre-processings (Stemming, Ultra-stemming and matrix generation) applied to the document NYT19990412.0403 from DUC 2006. Document is split in sentences; punctuation and case are removed; words are normalized.

For comparison, four methods of normalization were applied after filtering:

- Lemmatization by simple dictionary of morphological families: 1.32M words-entries in Spanish, 208K words in English and 316K in French.

- Porter's Stemming, available at Snowball site: `http://snowball.tartarus.org/texts/stemmersoverview.html`) for English, Spanish, French among other languages.

- Raw text without normalization.

- Ultra-stemming: the n first letters of each word. For example, in the case of Ultra-stemming of $n = 1$ (FIX_1), inflected verbs "sing", "song", "sings", "singing"... or proper names "smith", "snowboard", "sex",... are all replaced by letter "**s**".

3.3 Why ultra-stemming could work?

Although this technique could be considered a brutal destruction of the lexicon, Ultra-stemming is, in fact, an extreme stemming. That is, this truncation represents with minimum information, what we call the *stem of the stem*. In the case of Ultra-stemming with $n = 1$, the construction of the vectors-phrases is performed in a space of $j = 1, 2, ...$ 32 classes, which produces a dense vector representation.

Of course, classes are not equally populated. Figures 1 to 3 show the ranking of letters of three corpora in English, Spanish and French. The numbers and function words were previously removed.

In an automatic extractive summarizer, the weight of phrases is represented in a suitable vector space. However, if the representation is too large, the resulting representation is very sparse, which can difficult the weighting of the sentences. Two hypotheses are the basic ideas for using Ultra-stemming in automatic summarization task.

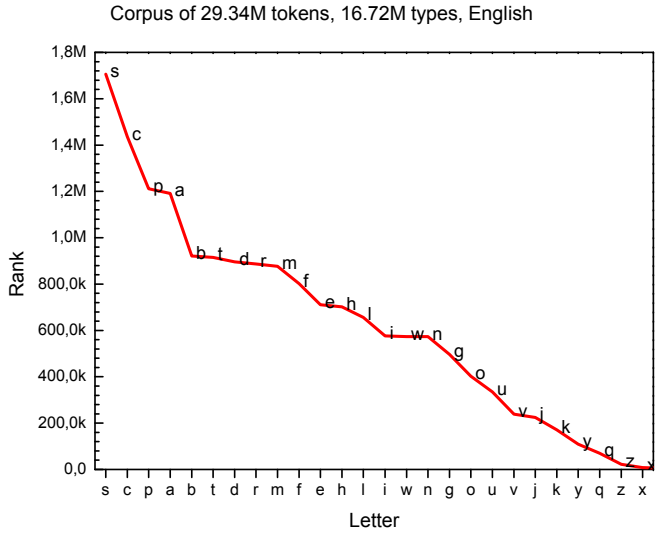

Figure 1: Scatter plot of first letter ranking for the English corpus. There are 16.72M of types, after filtering of functional words and punctuation.

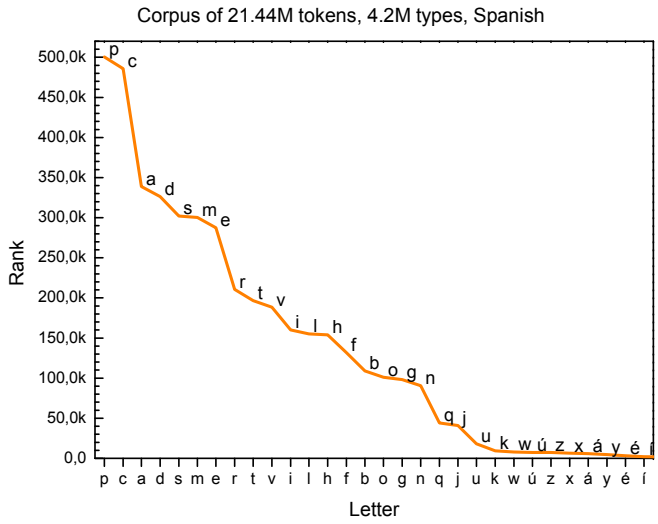

Figure 2: Scatter plot of first letter ranking for the Spanish corpus. There are 4.53M of types, after filtering of functional words and punctuation.

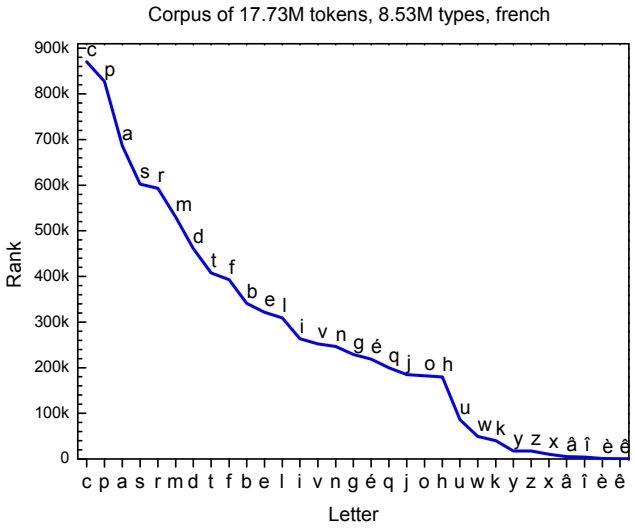

Figure 3: Scatter plot of first letter ranking for the French corpus. There are 8.53M of types, after filtering of functional words and punctuation.

The first hypothesis is that a more condensed, but retaining important information of the original representation, would enable a more effective weighting for phrases extraction. Ultra-stemming produces an extremely compact representation of documents, in a Vector Space that

can reach only thirty letters, using the representation of one letter per word. One way of evaluating the efficacity of a vector representation can be by calculating the density of the resulting matrix. This point will be discussed in detail in the next section. The other way is to show that two matrices A and B are equivalent in the sense that they contain a number of similar informations. If $A < B$, and A and B represent approximately the same information, then it may be preferable to use the representation given by A instead of B.

Now, how does one know that two matrices contain about the same information? The second hypothesis is that if the matrices A and B are correlated, then they probably represent similar information. This point will be proved in Section 4 by the Mantel statistic test.

3.4 Matrix density

Pre-processing and vectorization of documents will produce very sparse matrices. However the density of matrices generated is directly dependent on pre-processing algorithm used. Intuitively, the density of matrices generated by Ultra-stemming must be much greater than those generated by classical normalizations. We have calculated the density δ of a matrix $S_{[P\times N]}$, of P phrases and a vocabulary of N words as a fraction of occurrences C_w of the word w (elements other than 0), divided by the size of the matrix $\rho = P \times N$. The equation 1 calculates the density of S.

$$(1) \quad \delta(S) = \frac{C_w > 0}{\rho}$$

This density can be an indicator of the amount of information in relation to the volume of the matrix: lower density implies a greater amount of computation for ranking sentences. As shown in table 4, the matrix produced by Ultra-stemming of letters produces a higher average density on the studied corpora. The matrices generated by Ultra-stemming are filled approximately 50% (56% for English, 64% for Spanish and 47% for French). The volume of the matrix generated by each pre-processing method in relation to the size of the matrix in plain text, is given by:

$$(2) \quad V = \frac{\rho(\bullet)}{\rho(\text{Raw})}$$

This volume represents a small fraction (between 5% and 13% depending on the language) of the matrix equivalent of plain text.

In case of the corpus *Medicine Clínica* the standard matrices (lemm $\approx$ 101%, stem $\approx$ 103%) are slightly larger than the matrix produced by the plain text (raw). This can be explained by the presence of *Hapax legomenon*. In the case of plain text, a large number of *Hapax* ($f = 1$) is eliminated and it can produce matrices slightly smaller.

DUC'04		$\langle P \rangle = 238.0$	**Size**	**Volume V**
Pre-processing	**Density** δ	$\langle N \rangle$	$\rho = \langle P \rangle \times \langle N \rangle$	RAW=100%
LEMMATIZATION	2.6%	405.5	96 509.0	96.0%
STEMMING	2.4%	418.2	99 531.6	99.0%
RAW	2.3%	424.3	100 983.4	100.0%
FIX_1	**55.6%**	**25.6**	**6 092.8**	**6.0%**
Medicina Clínica		$\langle P \rangle = 88.6$	**Size**	**Volume V**
Pre-processing	**Density** δ	$\langle N \rangle$	$\rho = \langle P \rangle \times \langle N \rangle$	RAW=100%
LEMMATIZATION	5.9%	177.0	15 682.2	101.3%
STEMMING	5.7%	179.3	15 886.0	102.6%
RAW	5.1%	174.7	15 478.4	100.0%
FIX_1	**63.7%**	**22.2**	**1 966.9**	**12.7%**
Pistes		$\langle P \rangle = 319.7$	**Size**	**Volume V**
Pre-processing	**Density** δ	$\langle N \rangle$	$\rho = \langle P \rangle \times \langle N \rangle$	RAW=100%
LEMMATIZATION	2.0%	457.7	146 326,7	90.0%
STEMMING	1.9%	474.5	151 697.7	93.0%
RAW	1.6%	508.5	162 567.5	100.0%
FIX_1	**46.8%**	**25.0**	**7 992.5**	**4.9%**

Table 4: Matrix density for three corpora data. The mean dimension of matrix S, $\rho = \langle P \rangle \times \langle N \rangle$. Density $\delta(S)$ is calculated by equation 1 and Volume by equation 2.

Statistics for summarization DUC'04 English, *Medicina Clínica* Spanish and Pistes French corpora, after removing stop-words, *Hapax legomenon* and punctuation, are shown in table 5. The mode of letters per word is 5, 6 and 7, and 6 respectively for each language.

Corpus	**Words**	**Letters**	**Mean of letters per word**	**Mode on generic corpus**
DUC'04	11 956 sentences			**English**
LEMMATIZATION	137 454	800 723	5.83	•
STEMMING	137 101	764 015	5.57	•
RAW	136 582	902 914	6.61	**5**
FIX_1	137 461	137 461	1.00	•
Medicina Clínica	4 480 sentences			**Spanish**
LEMMATIZATION	56 063	484 281	8.64	•
STEMMING	56 067	410 048	7.31	•
RAW	56 115	526 660	9.38	**6-7**
FIX_1	56 347	56 347	1.00	•
Pistes	16 037 Sentences			**French**
LEMMATIZATION	167 056	1 505 169	9.01	•
STEMMING	167 231	1 264 774	7.56	•
RAW	167 677	1 589 190	9.48	**6**
FIX_1	168 329	168 329	1.00	•

Table 5: Statistics for three summarization corpora after filtering and removing punctuation.

Figures 4, 5 and 6 show the average distribution of letters per word by the three summary

corpora, after the filtering described in 3.2. Curves are shown normalized between $[0, 1]$ for the large generic and representative of the language corpora (cf Section 3.1) and the corpora used in each of the summaries experiments.

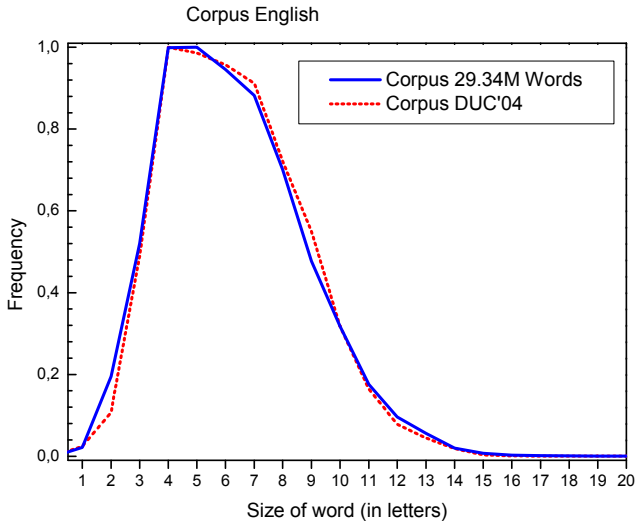

Figure 4: Scatter plot of mean length of words for two English corpora (heterogeneous and summarization raw corpora after filtering).

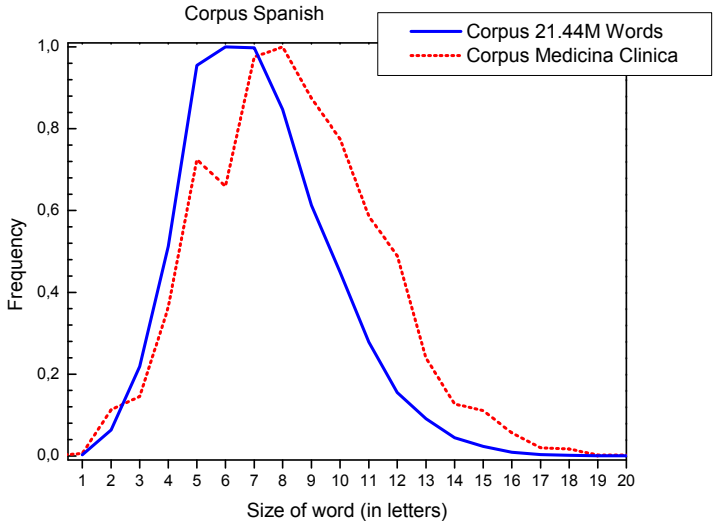

Figure 5: Scatter plot of mean length of words for two Spanish corpora (heterogeneous and summarization raw corpora after filtering).

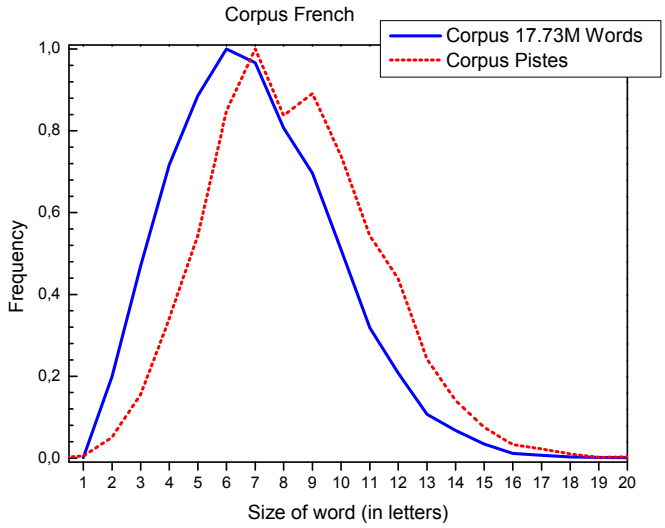

Figure 6: Scatter plot of mean length of words for two French corpora (heterogeneous and summarization raw corpora after filtering).

4 Matrix test correlation: the test of Mantel

Different methods of data analysis as ranking are based on distance matrices. [6] indicates: "In some cases, researchers may wish to compare several distance matrices with one another in order to test a hypothesis concerning a possible relationship between these matrices. However, this is not always evident. Usually, values in distance matrices are, in some way, correlated and therefore the usual assumption of independence between objects is violated in the classical tests approach. Furthermore, often, spurious correlations can be observed when comparing two distances matrices."

As [6] shows, in the Mantel test [33], the null hypothesis is that distances in a matrix A are independent of the distances, for the same objects, in another matrix B. In other words, we are testing the hypothesis that the process that has generated the data is or is not the same in the two sets. Then, testing of the null hypothesis is done by a randomization procedure in which the original value of the statistic is compared with the distribution found by randomly reallocating the order of the elements in one of the matrices. The measure used for the correlation between A and B is the Pearson correlation coefficient:

$$(3) \quad r(A,B) = \frac{1}{P-1} \sum_{i=1}^{P} \sum_{j=1}^{P} \left[\frac{A_{i,j} - \langle A \rangle}{\sigma_A} \right] \left[\frac{B_{i,j} - \langle B \rangle}{\sigma_B} \right]$$

where P is the number of elements in the lower (upper) triangular part of the matrix, $\langle A \rangle$ is mean for A elements and σ_A is the standard deviation of A elements.

Coefficient $r > 0$ measures the linear correlation and hence is subject to the same statistical assumptions. Consequently, if non-linear relationships between matrices exist, they will be degraded or lost ($r < 0$). The testing procedure for the simple Mantel test goes is the same of

[6], and it is as follows: Assume two symmetric dissimilarity matrices A and B of size $[P \times P]$. The rows and columns correspond to the same objects.

1. Compute the Pearson correlation coefficient $r(A, B)$ between the corresponding elements of the lower-triangular part of the A and B, using equation 3.

2. Permute randomly rows and the corresponding columns of the matrix A, creating a new matrix A'.

3. Compute $r(A', B)$ between matrices A' and B.

4. Repeat steps 2 and 3 a great number of times. This will constitute the reference distribution under the null hypothesis.

The calculation of the correlation between the matrix generated by the Ultra-stemming and others normalization methods is not straightforward, because the matrices are not square. In general, the matrix produced by the Ultra-stemming have a smaller number of columns than the other ones. Then, to calculate a correlation between matrices of different number of columns, each matrix must be converted in a symmetric matrix.

Let $S'_{[P \times N']}$ of P rows and N' columns be a matrix produced by Ultra-stemming, and let $S_{[P \times N]}$ of P rows and N columns, be a matrix produced by a classic method of normalization such that stemming, lemmatization, etc. We have the condition that: $N' \leq N$. Let the new matrices be $A_{[P \times P]} = S \times S'^T$ and $B_{[P \times P]} = S \times S'^T$. They are square symmetrical. A standard Mantel test can indicate the degree of similarity between A and B. If the similarity is high ($r > 0$) with a high confidence value ($p \to 0$), means that the information of the matrix A is substantially the same as that contained in the matrix B. In other words, we could replace S' for S, for purposes of a vector representation of documents.

Tables 6, 7 and 8 show the correlation of the Mantel test for the three summary corpora studied. The correlation was calculated between the matrices S generated by lemmatization (LEMM), stemming (STEM), plain text (RAW) and the matrix S' generated by Ultra-stemming FIX_1 using the initial letter. In all cases the correlation is positive with p-value < 0.001, which is significant. The calculations were performed with the `zt` program written in C, of Eric Bonnet and Yves Van de Peer[11] [5].

DUC'04	LEMM	STEM	RAW	FIX_1
LEMM	•	0.96149	0.91287	0.51904
STEM	0.96149	•	0.94492	0.53800
RAW	0.91287	0.94492	•	0.46914
FIX_1	0.51904	0.53800	0.46914	•

Table 6: Mantel test correlation for corpus DUC'04 data (English, p-value=0.001).

[11] zt: a software tool for simple and partial Mantel tests. This program can be downloaded from the Web site `http://bioinformatics.psb.ugent.be/software/details/ZT`

Medicina Clínica	LEMM	STEM	RAW	FIX_1
LEMM	•	0.96725	0,91174	0.58541
STEM	0.96725	•	0,91942	0,49614
RAW	0,91174	0,91942	•	0,51503
FIX_1	0,58541	0,49614	0,51503	•

Table 7: Mantel test correlation for corpus *Medicina Clínica* data (Spanish, p-value=0.001).

Pistes	LEMM	STEM	RAW	FIX_1
LEMM	•	0.93016	0.85708	0.53801
STEM	0.93016	•	0,89499	0.51641
RAW	0.85708	0,89499	•	0.45156
FIX_1	0,53801	0.51641	0.45156	•

Table 8: Mantel test correlation for corpus PISTES data (French, p-value=0.001).

According to these correlations, in DUC'04 English corpus, the method Fix_1 is more correlated with Stemming normalization. In Spanish and French corpora, Fix_1 seems slightly correlated with the model lemmatization. This is intuitively correct and according to the reduced variability of English in relation to Spanish or French.

5 Experiments

Ultra-stemming method described in the previous section has been implemented and evaluated in several corpora in English, Spanish and French languages. The following subsections present details of the different experiments.

5.1 Summarizers

Three summarization systems were used in our experiments: CORTEX, ENERTEX and ARTEX. All systems have used the same text representation based on Vector Space Model, described in Section 3.2.

- CORTEX is a single-document summarization system using several metrics and an optimal decision algorithm [43, 41].
- ENERTEX is summarization system based in Textual Energy concept [15]: text is represented as a spin system where spins $\uparrow$ represents words that their occurrences are $f > 0$ (spins $\downarrow$ if the word is not present).
- ARTEX (*AnotheR TEXt summarizer*) is a single-document summarization system that computes the score of a sentence by calculating a dot product between a sentence vector and a frequencies vector, multiply by lexical used.

We have conducted our experimentation with the following languages, summarization tasks, summarizers and data sets: 1) Generic multi-document-summarization in English with the corpus DUC'04; 2) Generic single-document summarization in Spanish with the corpus *Medicina Clínica* and 3) Generic single document summarization in French with the corpus PISTES.

Then, we have applied the summarization algorithms following the pre-processing algorithm and finally, results have been evaluated using FRESA.

5.2 Summaries Evaluation

To evaluate the quality of a summary is not an easy task, and remains an open question. DUC conferences have introduced the ROUGE evaluation [28], wich measures the overlap of n-grams between a candidate summary and reference summaries written by humans. In other hand, several metrics without references have been defined and experimented at DUC and TAC[12] workshops. FRESA measure [42] is similar to ROUGE evaluation but it does not uses reference summaries. It calculates the divergence of probabilities between the candidate summary and the document source. Among these metrics, Kullback-Leibler (KL) and Jensen-Shannon (JS) divergences have been used [29, 42] to evaluate the informativeness of summaries. In this paper, we use FRESA, based in KL divergence with Dirichlet smoothing, like in the 2010 and 2011 INEX edition [39], to evaluate the informative content of summaries by comparing their n-gram distributions with those from source documents.

FRESA only considered absolute log-diff between frequencies. Let T be the set of terms in the source. For every $t \in T$, we denote by C_t^T its occurrences in the source and by C_t^S its occurrences in the summary. The FRESA package computed the divergence between the source and the summaries as:

$$\mathcal{D}(T||S) = \sum_{t \in T} \left| \log\left(\frac{C_t^T}{|T|} + 1\right) - \log\left(\frac{C_t^S}{|S|} + 1\right) \right| \quad (4)$$

To evaluate the quality of generated summaries, several automatic measures were computed:

- FRESA_1: Unigrams of single stems after removing stop-words.
- FRESA_2: Bigrams of pairs of consecutive stems (in the same sentence).
- FRESA_{SU4}: Bigrams with 2-gaps also made of pairs of consecutive stems but allowing the insertion between them of a maximum of two stems.
- $\langle \text{FRESA} \rangle = \frac{\text{FRESA}_1 + \text{FRESA}_2 + \text{FRESA}_{SU4}}{3}$ is the mean of FRESA values, and represents the final score in our experiments.

The scores of FRESA are normalized between 0 and 1. High values mean less divergence regarding the source document summary, reflecting a greater amount of information content. All summaries produced by systems were evaluated automatically using FRESA package.

5.3 Results

Below we present separate results for the three languages. In this way, we have analyzed linguistic phenomena specific to each language.

5.3.1 English corpus

Results in figure 7 show that Ultra-steming improves the score of the three automatic summarizer systems. This result is remarkable for FIX_1, whose average matrix represents only 6% of the matrix volume in plain text.

[12] www.nist.gov/tac

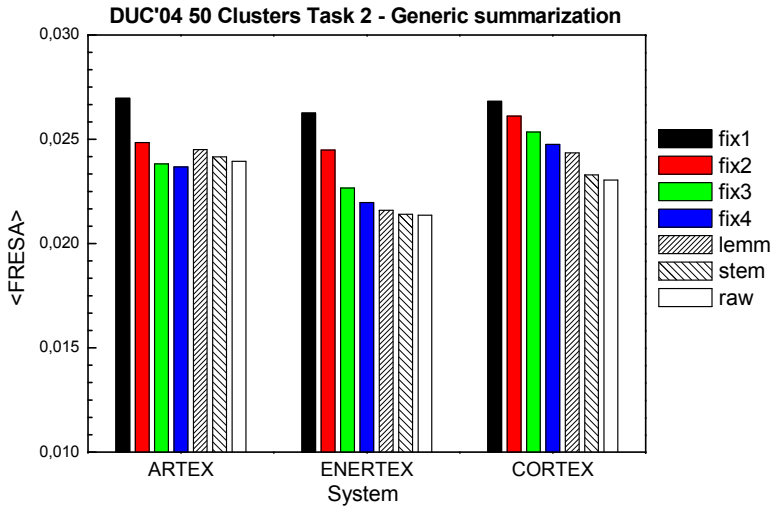

Figure 7: Histogram plot of content evaluation for corpus DUC 2004 Task 2, with ⟨FRESA⟩ measures, for each summarizer and each normalization.

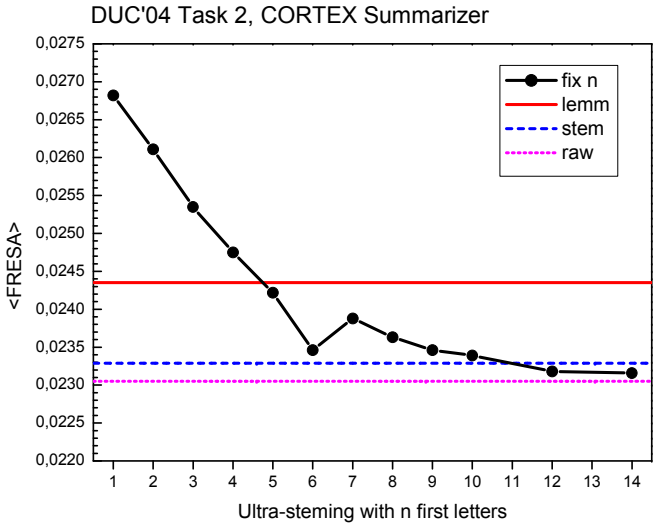

Figure 8: Scatter plot of ⟨FRESA⟩ mean of Ultra-stemming using n first letters (corpus DUC 2004 Task 2, CORTEX summarizer).

As shown in Figure 7, the performance of the three summarizers is improved using the Ultra-stemming in relation to other normalizations. So, in particular, using lemmatization (the best score between the two classic normalizations), the summarizer Artex, goes from 0.02451

to 0.02697 using normalization FIX$_1$, i.e. an increase of 10%. CORTEX increases of 0.02435 to 0.02682, an augmentation of 10.1% and summarizer ENERTEX increases of 0.02141 to 0.02626, an augmentation of 22.7%.

A detailed analysis for a particular summarizer is shown in Figure 8. This figure shows the average score FRESA obtained on DUC'04 English corpus, in function of Ultra-stemming used, of $n = 1, 2, ..., 14$ letters, for the automatic summarizer CORTEX. By comparison, the values FRESA for lemmatization (lemm), stemming (stem) and plain text (raw) are shown in the graph.

5.3.2 Spanish corpus

Spanish is a language with greater variability than English. Results in figure 9 shown that ultra-steming improves the score of the three systems of automatic summarization utilized. In the case of summarizers CORTEX and ARTEX, stemming and lemmatization substantially obtains the same scores, which does not occur with ENERTEX. However, comparing Ultra-stemming against stemming FIX$_1$, the three summarizers are benefiting of an increased score (ARTEX 5%, ENERTEX 5.25% and CORTEX 7.11 %).

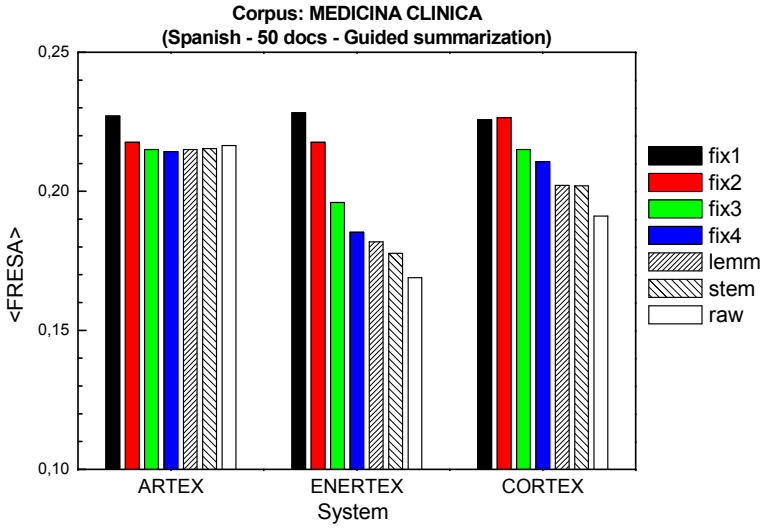

Figure 9: Histogram plot of content evaluation for Spanish corpus *Medicina Clínica* with ⟨FRESA⟩ scores for each summarizer.

Figure 10 shows the mean score ⟨FRESA⟩ on the Spanish corpus *Medicine Clínica*, based on the ultra-stemming ($n = 1, 2, ...14$ letters) using automatic summarizer CORTEX. Values FRESA for lemmatization (LEMM), stemming (STEM) and plain text (RAW) are also shown.

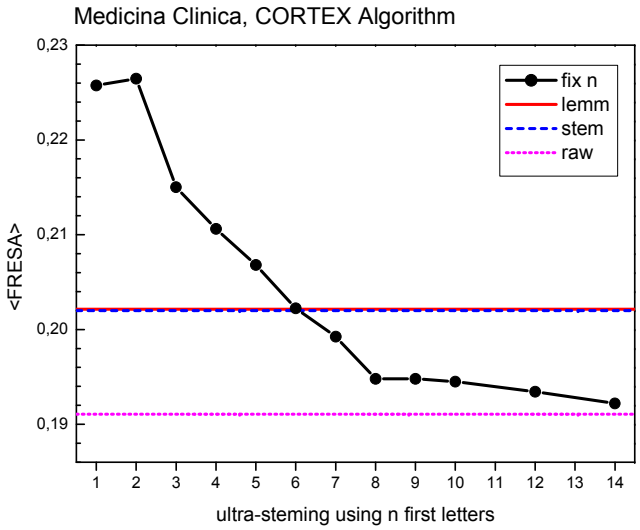

Figure 10: Scatter plot of ⟨FRESA⟩ mean vs. Ultra-stemming using n first letters (corpus *Medicina Clínica*, CORTEX summarizer).

5.3.3 French corpus

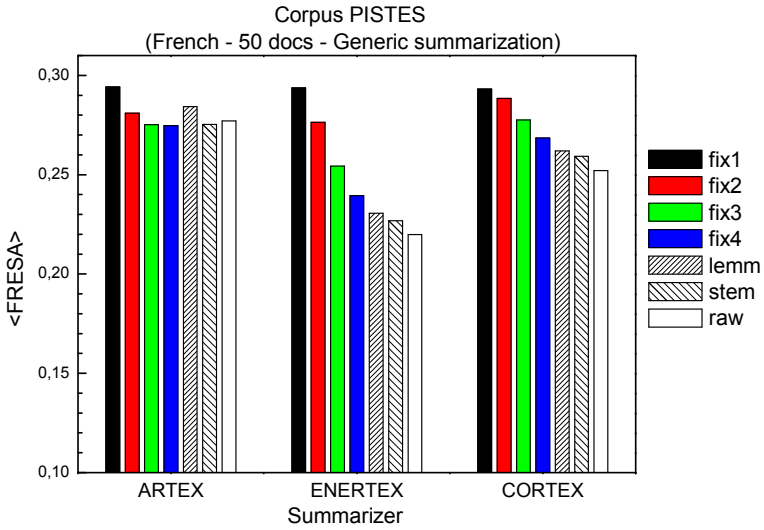

Figure 11: Histogram plot of content evaluation for French corpus PISTES with FRESA scores for each summarizer.

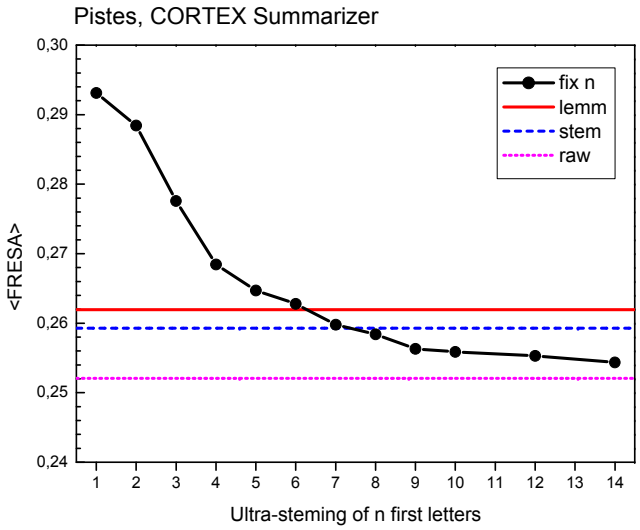

Figure 12: Scatter plot of ⟨FRESA⟩ mean vs. Ultra-stemming using n first letters (corpus PISTES, CORTEX summarizer).

Results in figure 11 show that Ultra-stemming improves the score of the three automatic summarization systems used. In particular, the summarizer ENERTEX using a stemming representation obtains a score FRESA of 0.25928, and using FIX_1 representation a score of 0.29311, i.e., an increase of more than 13%.

Finally, Figure 12 shows the detailed mean score ⟨FRESA⟩ on French corpus PISTES, as function of $n = 1, 2, ..., 14$ letters, using the automatic summarizer CORTEX. As well, it shows the values FRESA for lemmatization (LEMM), stemming (STEM) and plain text (RAW).

Overall for the three languages, beyond a certain number of letters (5 for English, 7 for the Spanish and 6 for French) Ultra-stemming loses its effectiveness and lemmatization score is higher. A view to the table 5 shows that this limit has a relationship with the mean, rather than the mode of letters per word in each language. Apparently, using Ultra-stemming is interesting when using a number of characters less than the mode of the language in question.

6 Discussion and conclusion

In this paper we have introduced and tested a simple pre-processing method suitable for automatic summarization text. Ultra-stemming is fast and simple. It reduces the size of the matrix representation, but it retains the information and charateristics of the document. An important aspect of our approach is that it does not requires linguistic knowledge or resources which makes it a simple and efficient pre-processing method to tackle the issue of Automatic Text Summarization.

And what about times ?

In general, the processing times of Ultra-stemming FIX_1 are shorter compared to all others methods. Of course, processing time depends of summarizer algorithm and pre-processing

algorithm. In general, processing time τ is function of:

$$\tau = \text{time(filtering)}+\text{time(normalization)}+\text{time(summarizer)}$$

In our experiments, time(filtering) is independent of the summarizers and generally, filtering algorithm is very fast. The time(normalization) depends on algorithm used (stemming, lemmatizaton) and/or extern resource (dictionary of lemmatization). The time(summarizer) is intrinsic to each summarizer system.

By example, CORTEX is a very fast summarizer with $O(\log \rho^2)$ (where $\rho = P \times N$), and processing times for STEMMING, RAW and FIX_1 are close. In other hand, ENERTEX summarizer has a complexity of $O(\rho^2)$, then it needs more time to process the same corpus. In this case, Ultra-stemming is a very interesting alternative to summarize long corpora. Table 9 shows processing times for each corpus, following the normalization method for CORTEX, ARTEX and ENERTEX summarizers. All times are measured in a 7.8 GB of RAM computer, Core i7-2640M CPU @ 2.80GHz $\times$ 4 processor, running under 32 bits GNU/Linux (Ubuntu Version 12.04).

Summarizer	**Corpus**			**Time**
Cortex	**DUC'04**	**Medicina Clínica**	**Pistes**	**Mean (All)**
LEMMATIZATION	0.80'	2.88'	1.13'	1.60'
STEMMING	0.40'	0.26'	0.53'	0.54'
RAW	0.33'	0.26'	0.41'	0.40'
FIX_1	0.31'	0.26'	0.38'	**0.32'**
Artex	**DUC'04**	**Medicina Clínica**	**Pistes**	**Mean (All)**
LEMMATIZATION	1.71'	3.10'	2.70'	2.50'
STEMMING	1.35'	0.40'	2.11'	1.29'
RAW	1.30'	0.38'	2.13'	1.27'
FIX_1	0.41'	0.28'	0.51'	**0.40'**
Enertex	**DUC'04**	**Medicina Clínica**	**Pistes**	**Mean (All)**
LEMMATIZATION	9.25'	3.38'	18.63'	10.42'
STEMMING	9.28'	0.75'	18.38'	9.47'
RAW	9.16'	0.73'	20.76'	10.22'
FIX_1	3.93'	0.46'	8.35'	**4.25'**

Table 9: Statistics of processing times (in minutes) of three summarizers over three corpora.

Clearly, the lemmatization of a large dictionary is the most time-consuming strategy. This is notable in the Spanish corpus, using a 1.3M dictionary entries. Lemmatization is at the same time, the strategy that produces the best results after the Ultra-stemming (Fix_n with $n = 1...4$ letters). In the case of Artex summarizer, the gain in time is dramatic, going from 2.50' using lemmatization to 0.40 using Fix_1, i.e. a gain of 625%. This gain is 500% for Cortex and 245% for Enertex.

From our point of view, the Ultra-stemming of n letters has three important advantages:

1. A reduction of the space and the calculation time of automatic summarization algorithms based on the vector space model.
2. Improving of summary content, when using $n <$ mode in letters per word of each language.
3. Applications on resource sparse languages. Typically π languages where no lemmatizers, stemmers or parsers, neither corpora nor native linguist available, the Ultra-stemming can be an attractive alternative for automatic document summarizers.

Summarization using the Ultra-stemming representation for sentence scoring, improve the identification of most relevant sentences from documents. The results obtained on corpora in English, Spanish and French prove that Ultra-stemming can achieve good results for content quality. Tests with other corpora (DUC evaluation campaigns, TAC, INEX, etc.) in mono-and multi-document guided by a subject, and π languages (Nahuatl, Maya, Somali, Interlingua, etc.) using content evaluation with or without reference summaries still in progress.

References

[1] E. Airio. Word normalization and decompounding in mono- and bilingual IR. *Information Retrieval*, 9(3):249–271, 2006.

[2] J. Atserias, B. Casas, E. Comelles, M. González, L. Padró, and M. Padró. FreeLing 1.3: Syntactic and semantic services in an open-source NLP library. In *fifth International Conference on Language Resources and Evaluation (LREC'06), ELRA*, 2006.

[3] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1999.

[4] D. Bernhard. Apprentissage non supervisé de familles morphologiques par classification ascendante hiérarchique. In *TALN'07*, volume 1, pages 367–376, 2006.

[5] Eric Bonnet and Yves Van de Peer. zt: a software tool for simple and partial Mantel tests. *Journal of Statistical software*, 7(10):1–12, 2002.

[6] Eric Bonnet and Yves Van de Peer. zt: A sofware tool for simple and partial mantel tests. *Journal of Statistical Software*, 7(10):1–12, 10 2002.

[7] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[8] M.T Cabré Castellví. Typology of neologisms: a complex task. *Alfa (São Paulo)*, 50(2):229–250, 2006.

[9] M. Creutz and K. Lagus. Unsupervised Discovery of Morphemes. In *6th Workshop of the ACL Special Interest Group in Computational Phonology (SIGPHON)*, pages 21–30, 2002.

[10] M. Creutz and K. Lagus. Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0. Technical Report A81, Publications in Computer and Information Science, Helsinki University of Technology, 2005.

[11] Harold Daumé III. *Practical structured learning techniques for natural language processing*. PhD thesis, Los Angeles, CA, 2006.

[12] D.P. Lyras and K.N. Sgarbas and N.D. Fakotakis. Using the Levenshtein Edit Distance for Automatic Lemmatization: A Case Study for Modern Greek and English. In *19th IEEE International Conference on Tools with Artificial Intelligence - (ICTAI'07)*, volume 2, pages 428–435, 2007.

[13] H. P. Edmundson. New Methods in Automatic Extraction. *Journal of the Association for Computing Machinery*, 16(2):264–285, 1969.

[14] F. Namer. Flemm: Un analyseur Flexionnel de Français à base de règles. In Christian Jacquemin, editor, *Traitement automatique des Langues pour la recherche d'information*, pages 523–547. Hermes, 2000.

[15] Silvia Fernández, Eric SanJuan, and Juan-Manuel Torres-Moreno. Textual Energy of Associative Memories: performants applications of Enertex algorithm in text summarization and topic segmentation. In *Proceedings of the Mexican International Conference on Artificial Intelligence (MICAI'07)*, pages 861–871, Aguascalientes, Mexique, 2007. Springer-Verlag.

[16] C.G. Figuerola, R. Gómez Díaz, and E. López de San Román. Stemming and n-grams in Spanish: An evaluation of their impact on information retrieval. *Journal of Information Science*, 26(6):461–467, 2000.

[17] A.F. Gelbukh, M. Alexandrov, and S.-Y. Han. Detecting inflection patterns in natural language by minimization of morphological model. In Sanfeliu A., Trinidad J.F.M., and Carrasco-Ochoa J.A.,

editors, *9th Iberoamerican Congress on Pattern Recognition (CIARP'04), Progress in Pattern Recognition, Image Analysis and Applications*, volume 3287, pages 432–438. Lecture Notes in Computer Science, Springer-Verlag, Berlin, 2004.

[18] J.A. Goldsmith. Unsupervised Learning of the Morphology of a Natural Language. *Computational Linguistics*, 27(2):153–198, 2001.

[19] N. Grabar and P. Zweigenbaum. Acquisition automatique de connaissances morphologiques sur le vocabulaire médical. In *TALN'99*, pages 175–184. Pascal Amsili, Ed., 1999.

[20] C. Hammarström. Unsupervised Learning of Morphology: Survey, Model, Algorithm and Experiments. Master's thesis, Department of Computer Science and Engineering, Chalmers University, 2007.

[21] H. Hammarström. A Naive Theory of Morphology and an Algorithm for Extraction. In R. Wicentowski and G. Kondrak, editors, *SIGPHON 2006: ACL Special Interest Group on Computational Phonology*, pages 79–88, 2006.

[22] S. Helmut. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, September 1994.

[23] V. Hollink, J. Kamps, C. Monz, and M. De Rijke. Monolingual Document Retrieval for European Languages. *Information Retrieval*, 7(1-2):33–52, January 2004.

[24] C. Jacquemin and E. Tzoukermann. NLP for term variant extraction: synergy between morphology, lexicon and syntax. In Tomek Strzalkowski, editor, *Natural Language Information Retrieval*, volume 7 of *Text, Speech and Language Technology*, pages 25–74. Kluwer Academic Publishers, Dordrecht/Boston/London, 1999.

[25] T. Korenius, J. Laurikkala, K. Jarvelin, and M. Juhola. Stemming and lemmatization in the clustering of finnish text documents. In *CIKM'04: Thirteenth ACM Conference on Information and Knowledge Management*, pages 625–633. ACM Press, 2004.

[26] J. Kupiec, J. Pedersen, and F. Chen. A trainable document summarizer. In *Proceedings of the 18th Conference ACM Special Interest Group on Information Retrieval (SIGIR'95)*, pages 68–73, Seattle, WA, Etats-Unis, 1995. ACM Press, New York.

[27] Y. Lepage. Solving analogies on words: an algorithm. In *COLING-ACL'98*, pages 728–735, 1998.

[28] Chin-Yew Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In Marie-Francine Moens and Stan Szpakowicz, editors, *Proceedings of the Workshop Text Summarization Branches Out (ACL'04)*, pages 74–81, Barcelone, Espagne, july 2004. ACL.

[29] Annie Louis and Ani Nenkova. Automatic Summary Evaluation without Human Models. In *First Text Analysis Conference (TAC'08)*, Gaithersburg, MD, Etats-Unis, 17-19 November 2008.

[30] H.P. Luhn. The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958.

[31] I. Mani and M. Mayburi. *Advances in Automatic Text Summarization.* MIT Press, Cambridge, 1999.

[32] C.D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing.* The MIT Press, 1999.

[33] Nathan Mantel and Ranchhodbhai S. Valand. A Technique of Nonparametric Multivariate Analysis. *Biometrics*, 26(3):547–558, Sep., 1970.

[34] Juan Manuel Torres Moreno. Reagrupamiento en familias y lexematización automática independientes del idioma. *Revista Iberoamericana de Inteligencia Artificial*, 14(47):38–53, 2010.

[35] C.D. Paice. Another stemmer. *SIGIR Forum*, 24(3):56–61, 1990.

[36] C.D. Paice. Method for Evaluation of Stemming Algorithms Based on Error Counting. *Journal of the American Society for Information Science*, 47(8):632–649, 1996.

[37] M.F. Porter. An algorithm for suffix stripping. *Program*, 40(3):211–218, 2006.

[38] J. Ross Quinlan. *C4.5: Programs for Machine Learning (Morgan Kaufmann Series in Machine Learning).* Morgan Kaufmann, 1 edition, 1993.

[39] Eric SanJuan, Patrice Bellot, Véronique Moriceau, and Xavier Tannier. Overview of the inex 2010 question answering track (qa@inex). In Shlomo Geva, Jaap Kamps, Ralf Schenkel, and Andrew Trotman, editors, *Comparative Evaluation of Focused Retrieval*, volume 6932 of *Lecture Notes in Computer Science*, pages 269–281. Springer Berlin / Heidelberg, 2011.

[40] Simone Teufel and Marc Moens. Sentence extraction as a classification task. In I. Mani and M. Maybury, editors, *Proceedings of the ACL/EACL'97 Workshop on Intelligent Scalable Text Summarization*, Madrid, Espagne, 11 juillet 1997.

[41] Juan-Manuel Torres-Moreno. *Résumé automatique de documents: une approche statistique.* Hermès-Lavoisier, Paris, 2011.

[42] Juan-Manuel Torres-Moreno, Horacio Saggion, Iria da Cunha, and Eric SanJuan. Summary Evaluation With and Without References. *Polibits: Research journal on Computer science and computer engineering with applications*, 42:13–19, 2010.

[43] Juan-Manuel Torres-Moreno, Patricia Velázquez-Morales, and Jean-Guy Meunier. Cortex : un algorithme pour la condensation automatique des textes. In *Proceedings of the Conference de l'Association pour la Recherche Cognitive*, volume 2, pages 365–366, Lyon, France, 2001.

[44] A. Medina Urrea. Automatic Discovery of Affixes by means of a Corpus: A Catalog of Spanish Affixes. *Journal of Quantitative Linguistics*, 7(2):97–114, 2000.

[45] J. Vilares, M. A. Alonso, and M. Vilares. Extraction of complex index terms in non-English IR: A shallow parsing based approach. *Information Processing and Management*, 44(4):1517–1537, 2008.

[46] J. Vilares, D. Cabrero, and M. A. Alonso. Applying productive derivational morphology to term indexing of Spanish texts. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing (CICLing'01)*, volume 2004 of *Lecture Notes in Computer Science*, pages 336–348. Springer-Verlag, Berlin-Heidelberg-New York, 2001.